PCS 2428 / PCS 2059
Inteligência Artificial

Prof. Dr. Jaime Simão Sichman
Prof. Dra. Anna Helena Reali Costa

Agentes Inteligentes

## Problema: Auxílio a Compras na Web

2

## Problema: Automatização de sistemas de potência

objetos: rios, barragens, turbinas, transformadores, linhas, ...

3

## Problema: Produção de histórias interativas

. Criar ilusão da vida (ex. Walt Disney)

. Permitir interação com usuário

. Modelar comportamento e personalidade (ex. tamagotchi)

4

## Problema: navegação autônoma (ALVINN)

Velocidades → 100km/h

Distâncias → 140km

Usa RN associada a imagens para guiar uma van em rodovias públicas.

5

## Problema: exploração planetária

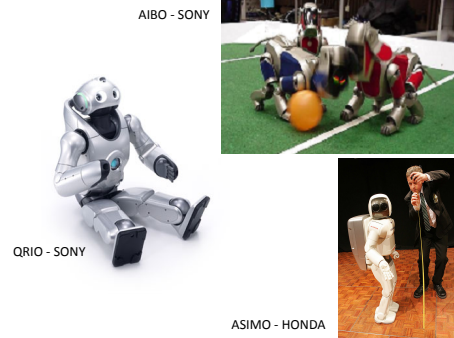
6

## Problema: aspirador de pó automático

2005: 2 milhões de unidades vendidas para uso doméstico

7

## Problema: robôs de estimação

AIBO - SONY

QRIO - SONY

ASIMO - HONDA

8

## E aí?

- O que estes problemas têm em comum?
  - Grande complexidade (número, variedade e natureza das tarefas)
  - Não há "solução algorítmica", mas existe conhecimento
  - Modelagem do comportamento de um ser inteligente (autonomia, aprendizagem, conhecimento, etc.)

9

## *O que é "ser inteligente"?*

- Ser inteligente é atuar como humanos.
- Ser inteligente é "pensar" como humanos.
- Ser inteligente é "pensar" racionalmente.
- Ser inteligente é atuar racionalmente.

*Racionalidade = capacidade de alcançar o sucesso esperado na execução de uma tarefa.*
*Grau de Sucesso: medida de desempenho a ser maximizada.*

10

- Inteligência Artificial (IA)
  - Há 50 anos lida com esses problemas.
  - **Objetivo**: construir (e aprender a construir) programas que, segundo critérios definidos, exibem um comportamento inteligente na realização de uma dada tarefa.

11

## Um programa de IA pode ser visto como um Agente Racional

- Plano da aula
  - O que é um Agente Racional (inteligente)?
  - Ambientes e Arquiteturas
  - Aplicações
  - Estado atual do conceito de agente

12

## Medida de Desempenho

- Critério que define o grau de sucesso de um agente na realização de uma dada tarefa
  O quê avaliar, Como avaliar, Quando avaliar
  - Esta medida deve ser imposta do exterior
  - Má escolha da MD pode acarretar comportamento indesejado
  - Compromissos entre objetivos múltiplos conflitantes
  - Resta o problema de saber **quando** avaliar o desempenho

14

## Agente Racional

- Agente Racional:
  - "Para cada seqüência perceptual possível, o agente racional deve **selecionar uma ação** que ele espera que maximize sua medida de desempenho, segundo a evidência dada pela **seqüência perceptiva** e os eventuais **conhecimentos** que tenha".
- Limitações de:
  - Sensores, atuadores, "raciocinador" (conhecimento, tempo, etc.)
- Agente racional deve ser autônomo:
  - Ter capacidade de raciocínio, decisão e de adaptação a situações novas, para as quais não foi fornecido todo o conhecimento necessário com antecedência

15

## A metáfora de agente decompõe:

1) Problema em:
   - percepções, ações, objetivos e ambiente (e outros agentes)
2) Tipo de conhecimento em:
   - Quais são as propriedades relevantes do mundo
   - Como o mundo evolui
   - Como identificar os estados desejáveis do mundo
   - Como interpretar suas percepções
   - Quais as conseqüências de suas ações no mundo
   - Como medir o sucesso de suas ações
   - Como avaliar seus próprios conhecimentos

16

## Agente de polícia

Agente

*raciocínio*

Conhecimento:
- leis
- comportamento dos indivíduos,...

Objetivo:
- fazer com que as leis sejam respeitadas

Ações:
- multar
- apitar
- parar, ...

*percepção*

*execução*

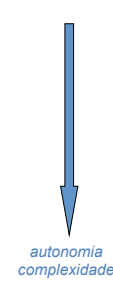
Ambiente

17

## Propriedades do Ambiente

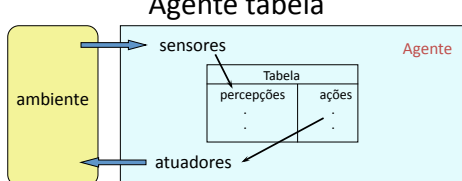
- totalmente observável x parcialmente observável
- determinístico x estocástico (envolve previsibilidade do próximo estado)
- episódico x seqüencial (envolve ter ou não conseqüências futuras decorrentes da decisão atual)
- estático x dinâmico (envolve tempo)
- discreto x contínuo (aplicado a estado, tempo, ações e/ou percepções)
- único agente x multiagente (envolve comunicação, cooperação, competição..)

18

## Estrutura do Agente

Agente = arquitetura de HW
+
arquitetura de SW

- Arquitetura de HW:
  – onde o agente vai ser implementado (dispositivo computacional, sensores e atuadores)
- Arquitetura de SW:
  – "arquitetura do agente": módulos básicos do programa e suas inter-relações

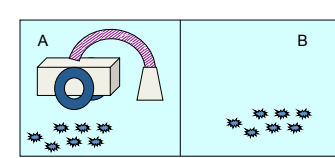
19

## Arquiteturas

- Agente tabela
- Agente reativo
- Agente baseado em modelo
- Agente baseado em objetivos
- Agente baseado em utilidade
- Agente aprendiz

*autonomia*
*complexidade*

20

## Agente tabela

ambiente → sensores; Agente; Tabela (percepções | ações); atuadores → ambiente

- Limitações
  – Mesmo para problemas simples ➔ tabelas muito grandes
    - ex. xadrez 30^100
  – Nem sempre é possível, por ignorância ou questão de tempo, construir a tabela
  – Não há autonomia nem flexibilidade
- Ambientes
  – observável, determinístico, episódico, estático, discreto (e minúsculo!)

21

A B

| PERCEPÇÃO [sala, estado] | AÇÃO |
|---|---|
| [A, limpo] | Ir para a direita |
| [A, sujo] | Aspirar |
| [B, limpo] | Ir para a esquerda |
| [B, sujo] | Aspirar |

22

```
function Agente-Tabela-Aspirador (percept) return uma ação

static: percepts – uma seqüência, inicialmente vazia
        table – uma tabela de ações indexada pela seqüência
                perceptiva, inicialmente totalmente especificada

append percept ao final de percepts
ação ← Lookup (percepts, table)
return ação
```

Uso MUITO limitado (impossível): seja P o conjunto de percepções possíveis e T, o tempo de vida do agente ➔ entradas da tabela = $\Sigma_{t=1..T} |P|^t$

➔ Como IA soluciona isso?

23

## Arquiteturas

- Agente tabela
- Agente reativo
- Agente baseado em modelo
- Agente baseado em objetivos
- Agente baseado em utilidade
- Agente aprendiz

24

## Agente reativo

- Vantagens e desvantagens
  - Regras condição-ação: representação inteligível, modular e eficiente
    - ex. **Se** velocidade > 60 **então** multar
  - Não pode armazenar uma <u>seqüência</u> perceptiva, pouca autonomia
- Ambientes:
  - Reflexo imprescindível em ambientes dinâmicos
  - Observável, episódico, pequeno

25

```
function Agente-Reativo-Aspirador ([local, status]) return uma ação

  if status=Sujo then return Aspira
  else if local=A then return Direita
  else if local=B then return Esquerda
```

OBS: utiliza a percepção corrente e ignora a história perceptual.

26

```
function Agente-Reflexo-Simples (percept) return uma ação

static: regras – um conjunto de regras condição-ação

estado ← Interpreta-Entrada(percept)
regra  ← Acha-Regra(estado, regras)
ação   ← Regra-Ação [regra]
return ação
```

Uso limitado: o ambiente tem que ser totalmente observável, pois o agente só funciona apropriadamente se a regra correta for disparada, o que depende da percepção atual realizada.

27

## Arquiteturas

- Agente tabela
- Agente reativo
- Agente baseado em modelo
- Agente baseado em objetivos
- Agente baseado em utilidade
- Agente aprendiz

28

## Necessidade de um modelo (1)

- Um agente puramente reativo que tenha observação parcial do ambiente pode cair em *deadlocks* ou *loops* infinitos.
- A forma mais efetiva de lidar com observabilidade parcial é "<u>lembrar</u> e <u>imaginar</u> a parte do mundo que não está observável no momento", i.e, manter um <u>estado interno</u> que dependa da história perceptual passada e reflita (ao menos em parte) aspectos não observados no estado atual.

29

## Necessidade de um modelo (2)

- Neste novo agente, para determinar como o mundo está num determinado momento, ele usa:
  - informações perceptuais atuais (como o agente reativo)
  - seu estado interno
  - informações a respeito de como o mundo evolui, independentemente de suas ações **(modelo do mundo)**
  - informações a respeito do impacto/efeito de suas próprias ações no mundo

**e, com isso, atualiza seu estado interno.**

30

## Agente baseado em modelo

- Desvantagem: pouca autonomia
  - não tem objetivo, não encadeia regras
- Ambientes: determinístico e pequeno
  - pode lidar com a observabilidade parcial

Ex.: carro mudando de faixa, no trânsito (percepção parcial, precisa do modelo do mundo pra prever onde estão os outros carros).



function Agente-Com-Estado (*percept*) return uma ação

static: *estado* – uma descrição do estado corrente do mundo
*regras* – um conjunto de regras condição-ação
*ação* – a ação mais recente, inicialmente nula

*estado* ← Atualiza-Estado(*estado, ação, percept*)
regra ← Acha-Regra(*estado*, *regras*)
*ação* ← Regra-Ação [regra]
return *ação*

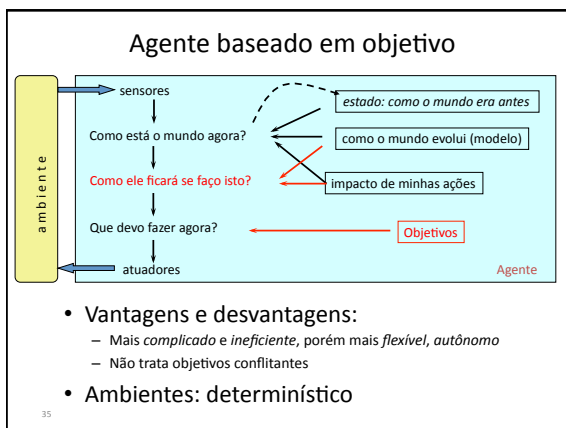


## Arquiteturas

- Agente tabela
- Agente reativo
- Agente baseado em modelo
- Agente baseado em objetivos
- Agente baseado em utilidade
- Agente aprendiz



## Necessidade de metas/objetivos

- Além do estado interno, um agente precisa de alguma informação a respeito de metas, indicando situações desejáveis, para decidir a melhor ação a executar.
- Assim, pode combinar as informações do impacto de suas ações com seus objetivos, de modo a fazer considerações acerca do futuro (predições) e decidir melhor suas ações.
  - O agente poderá ter que considerar longas seqüências de ações encadeadas para poder atingir sua meta ➔ **busca** e **planejamento** são subáreas de IA que visam determinar a **seqüência de ações** que leva o agente ao objetivo.



## Agente baseado em objetivo

- Vantagens e desvantagens:
  - Mais *complicado* e *ineficiente*, porém mais *flexível*, *autônomo*
  - Não trata objetivos conflitantes
- Ambientes: determinístico



## Arquiteturas

- Agente tabela
- Agente reativo
- Agente baseado em modelo
- Agente baseado em objetivos
- Agente baseado em utilidade
- Agente aprendiz

## Medida de desempenho mais geral: função de utilidade

- Uma função de utilidade mapeia um estado (ou seqüência de estados) em um número real que descreve o grau de satisfação associado ao estado.
- Permite decisões racionais quando há:
  - Objetivos conflitantes (ex: velocidade e segurança) ➔ neste caso a função de utilidade define um compromisso adequado entre eles.
  - Múltiplos objetivos, nenhum dos quais se tem certeza de ser atingido ➔ a função de utilidade permite um balanceamento entre a possibilidade de sucesso com a importância de cada objetivo.

37

## Agente baseado em utilidade

- Ambiente: sem restrição
- Desvantagem: não tem adaptabilidade (não aprende)

38

## Arquiteturas

- Agente tabela
- Agente reativo
- Agente baseado em modelo
- Agente baseado em objetivos
- Agente baseado em utilidade
- Agente aprendiz

39

## Agente que aprende

- Ambiente: sem restrição
- Vantagem: tem adaptabilidade (aprende)

40

## Simulação do ambiente

- Às vezes é mais conveniente simular o ambiente
  - mais simples
  - permite testes prévios
  - evita riscos, etc...
- O ambiente (programa)
  - recebe os agentes como entrada
  - fornece repetidamente a cada um deles as percepções corretas e recebe as ações
  - atualiza os dados do ambiente em função dessas ações e de outros processos (ex. dia-noite)
  - é definido por um estado inicial e uma função de atualização
  - deve refletir a realidade

41

## Simulação de ambiente

```
função simulaAmbiente (estado, funçãoAtualização,agentes,final)
    repita

      para cada agente em agentes faça
            Percept[agente] := pegaPercepção(agente,estado)

      para cada agente em agentes faça
            Action[agente] := Programa[agente] (Percept[agente])

       estado := funçãoAtualização(ações, agentes, estado)

       scores := avaliaDesempenho(scores,agente,estado) //opcional

    até final
```

42

## Desenvolver agentes inteligentes

- Projeto:
  - Modelar tarefa em termos de ambiente, percepções, ações, objetivos e utilidade
  - Identificar o tipo de ambiente
  - Identificar a arquitetura de agente adequada ao ambiente e tarefa
- Implementação
  - O gerador e o simulador de ambientes
  - Componentes do agente (vários tipos de conhecimento)
  - Testar o desempenho com diferentes instâncias do ambiente

43

## Evolução da noção de agente além das fronteiras da IA....

- Agentes em IA
  - Metodologia (metáfora) para projeto de sistemas
  - Sistemas multiagentes e robótica
- Agentes em Computação
  - **Adoção de uma nova metáfora** (antropomórfica e sociológica). Extrapolação de OOP
  - Integração de técnicas de IA
  - Novas tecnologias próprias à Web (ex. mobilidade)
  - Marketing (moda)
- Agentes: técnica ou metodologia ?

44